# Require-Ensure Assertions as preconditions and postconditions

CS 152 Lecture

R. K. Joshi
IIT Bombay

# What to check in Preconditions (*Require* Clause)?

- **The inputs**

If the language enforces typing rules, we don't have to worry about values of *wrong* types

But all values of that type are not acceptable, in other words, the function is a partial function as in div(int a, int b)

- **The Current State**

When we are working on a **stateful** object

# What to check in Postcondition (*Ensure* Clause)?

- **The output**

•is the output satisfying the contract?

- **The Current State**

Is the current state correct as per the contract?

# Example: Require and Ensure

## There is still a problem

```
#include<iostream>
#include <string>
#include <cassert>
using namespace std;

#define require assert
#define ensure assert

bool allnumeric(string s) {

   for (int i=0; i<s.size(); i++){
        if (s[i]<48) return false;
        if (s[i]>58) return false;
   }
   return true;
};
```

```
int str2int (string s) {
        require (s.size() > 0);


        int i;
        for (int i=0; i<s.size(); i++) {
                i = i + s[i] -48;
        }
        return i;
};
int main() {

        cout << str2int ("1234")<< endl;
        cout << str2int ("0012") << endl;
        cout << str2int ("hallo") << endl;
        cout << str2int ("attitude") << endl;
}
```

# What can we add into assert, ensure, require clauses

- conditions that return a boolean value
  - On input parameters
  - Using current state
- The condition may be 'embedded' inside a function or a procedure, if the assertion logic is complex
  - One such Assertive Function is used in the next slide

# Example: Require and Ensure

## There is still a problem

```
#include<iostream>
#include <string>
#include <cassert>
using namespace std;

#define require assert
#define ensure assert

bool allnumeric(string s) {

   for (int i=0; i<s.size(); i++){
        if (s[i]<48) return false;
        if (s[i]>58) return false;
   }
   return true;
};
```

```
int str2int (string s) {
        require (s.size() > 0);
        require (allnumeric(s));

        int i;
        for (int i=0; i<s.size(); i++) {
                i = i + s[i] -48;
        }
        return i;
};
int main() {

        cout << str2int ("1234")<< endl;
        cout << str2int ("0012") << endl;
        cout << str2int ("hallo") << endl;
        cout << str2int ("attitude") << endl;
}
```

# Example: Require and Ensure

## There is still a problem

```
#include<iostream>
#include <string>
#include <cassert>
using namespace std;

#define require assert
#define ensure assert

bool allnumeric(string s) {

   for (int i=0; i<s.size(); i++){
        if (s[i]<48) return false;
        if (s[i]>58) return false;
   }
   return true;
};
```

```
int str2int (string s) {
        require (s.size() > 0);
        require (allnumeric(s));

        int i;
        for (int i=0; i<s.size(); i++) {
                i = i + s[i] -48;
        }
        ensure (i>0);
        return i;
};
int main() {

        cout << str2int ("1234")<< endl;
        cout << str2int ("0012") << endl;
        cout << str2int ("hallo") << endl;
        cout << str2int ("attitude") << endl;
}
```

# Require and Ensure

```
#include<iostream>
#include <string>
#include <cassert>
using namespace std;

#define require assert
#define ensure assert

bool allnumeric(string s) {

   for (int i=0; i<s.size(); i++){
        if (s[i]<48) return false;
        if (s[i]>58) return false;
   }
   return true;
};
```

```
int str2int (string s) {
        require (s.size() > 0);
        require (allnumeric(s));

        int x;
        for (int i=0; i<s.size(); i++) {
                x = x + s[i] -48;
        }
        ensure (x>0);
        return i;
};
int main() {

        cout << str2int ("1234")<< endl;
        cout << str2int ("0012") << endl;
        cout << str2int ("hallo") << endl;
        cout << str2int ("attitude") << endl;
}
```

# Require and Ensure – Are we providing the full specification?

```
#include<iostream>
#include <string>
#include <cassert>
using namespace std;

#define require assert
#define ensure assert

bool allnumeric(string s) {

   for (int i=0; i<s.size(); i++){
        if (s[i]<48) return false;
        if (s[i]>58) return false;
   }
   return true;
};
```

```
int str2int (string s) {
        require (s.size() > 0);
        require (allnumeric(s));

        int x = 1;   // satisfies ensure!

        ensure (x>0);
        return i;
};
int main() {

        cout << str2int ("1234")<< endl;
        cout << str2int ("0012") << endl;
        cout << str2int ("hallo") << endl;
        cout << str2int ("attitude") << endl;
}
```